

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of:

Peter WILLIAMSON, et al.

Serial No.: 10/767,847

Filed: January 30, 2004

For: IMPLEMENTING HANDWRITTEN
SHORTHAND IN A COMPUTER
SYSTEM

Atty. Docket No.: 003797.00737

Group Art Unit: 2178

Examiner: Gregory J. Vaughn

Confirmation No.: 4097

APPEAL BRIEF

U.S. Patent and Trademark Office
Customer Service Window
Mail Stop - Appeal
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Sir:

This is an Appeal Brief filed in support of Appellants' March 19, 2007, Notice of Appeal. Appeal is taken from the final Office Action mailed October 19, 2006 (hereafter, "Final Office Action") and the Advisory Action mailed January 23, 2007. Please charge any fees to our Deposit Account No. 19-0733. In addition, any extension of time necessary for acceptance of this paper is hereby requested.

REAL PARTY IN INTEREST

37 C.F.R. § 41.37(c)(1)(i)

The owner of this application, and the real party in interest, is Microsoft Corporation.

RELATED APPEALS AND INTERFERENCES

37 C.F.R. § 41.37(c)(1)(ii)

There are no related appeals or interferences.

STATUS OF CLAIMS

37 C.F.R. § 41.37(c)(1)(iii)

Claims 1, 13-15, 17, 19-21, 24-27, and 34-39 are pending. Of these, claims 1, 13-15, 17, 19, 20, 24-27, and 34-36 stand rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 6,493,464 to Hawkins et al. (hereinafter “Hawkins”), and claims 21 and 37-39 stand rejected under 35 U.S.C. § 103 as being unpatentable over Hawkins.

Appellants hereby appeal the rejection of claims 1, 13-15, 17, 19-21, 24-27, and 34-39.

STATUS OF AMENDMENTS

37 C.F.R. § 41.37(c)(1)(iv)

The most recent Amendment was filed July 28, 2006, which has been entered. No subsequent Amendments have been filed.

SUMMARY OF CLAIMED SUBJECT MATTER

37 C.F.R. § 41.37(c)(1)(v)

In making reference herein to various embodiments in the specification text and drawings to explain the claimed invention, Appellant does not intend to limit the claims to those embodiments; all references to the specification and drawings are illustrative unless otherwise explicitly stated.

Over the past few years, it has become more common for hand-held and laptop computers to incorporate stylus-based user interfaces. *Specification*, para. 02. Such stylus based computers typically include a digitizing flat panel display surface. *Id.* Using the tip of the stylus, a user may select, highlight, and write on the digitizing display surface. *Id.* The stylus-based computer interprets marks made using the stylus in order to manipulate data, enter text, and execute conventional computer application tasks such as spreadsheets, word processing programs, and the like. *Id.* Stylus-based computer systems can also recognize handwritten symbols, such as alphanumeric characters, as well as command-type gestures. *Id.*

The invention of claim 1 is directed to a method in a computer system. *Id.*, paras. 19-30, Fig. 1, Fig. 2, element 201, Fig. 10, Fig. 12. The method includes receiving user input identifying a symbol, a text expansion, and a program. *Id.*, paras. 35-37, paras. 40-42, para. 44, para. 45, Fig. 4, element 400, Fig. 5, elements 501, 502, 503, Fig. 6, element 400, Fig. 7,

elements 701, 702, 703, Fig. 8, element 400, Fig. 9, elements 901, 902, 903. The method further includes associating the text expansion and the program with the symbol. *Id.*, paras. 40-42, para. 44, para. 45, Fig. 5, elements 501, 503, Fig. 9, elements 901, 903. The method further includes receiving handwritten user input. *Id.*, para. 47, para. 53, Fig. 10, step 1001, Fig. 12, step 1201. The method further includes determining whether the handwritten user input represents the symbol. *Id.*, para. 47, para. 48, para. 53, Fig. 10, step 1002, Fig. 12, step 1202. The method further includes determining a context in which the handwritten user input is written. *Id.*, para. 46, para. 47, para. 52, para. 53, Fig. 10, step 1003, Fig. 12, step 1203. The method further includes choosing between either the text expansion or the program depending upon the determined context. *Id.*, paras. 46-49, para. 52, Fig. 10, step 1005, Fig. 12, step 1206. The method further includes either displaying the text expansion or launching the program depending upon the outcome of the choosing step. *Id.*, para. 48, para. 49, para. 52, Fig. 10, steps 1009, 1007, Fig. 12, steps 1208, 1209.

The invention of claim 14 is directed to a method in a computer system. *Id.*, paras. 19-30, Fig. 1, Fig. 2, element 201, Fig. 10, Fig. 12. The method includes receiving first handwritten user input including at least first handwritten user input, the first handwritten input being associated with both expanded text and a program. *Id.*, paras. 40-42, para. 44, para. 45, para. 47, para. 53, Fig. 5, elements 501, 503, Fig. 9, elements 901, 903, Fig. 10, step 1001, Fig. 12, step 1201. The method further includes determining whether the first handwritten user input includes second handwritten user input in addition to the first handwritten user input. *Id.*, para. 50, para. 51, para. 53, Fig. 10, step 1006, Fig. 12, step 1205. The method further includes choosing between either the expanded text or the program depending upon whether the first handwritten user input includes the second handwritten user input. *Id.*, para. 50, para. 51. The method further includes the step performed in response to the first handwritten user input, either displaying the expanded text or launching the program, depending on the outcome of the choosing step. *Id.*, para. 50, para. 51, para. 52, Fig. 10, steps 1009, 1007, Fig. 12, steps 1208, 1209.

The invention of claim 27 is directed to a method in a computer system. *Id.*, paras. 19-30, Fig. 1, Fig. 2, element 201, Fig. 10, Fig. 12. The method includes receiving handwritten user input. *Id.*, para. 47, para. 53, Fig. 10, step 1001, Fig. 12, step 1201. The method further includes recognizing the handwritten user input to determine a symbol. *Id.*, para. 47, para. 48, para. 53,

Fig. 10, step 1002, Fig. 12, step 1202. The method further includes determining expanded text represented by the symbol. *Id.*, para. 46. The method further includes determining a program represented by the symbol. *Id.* The method further includes either displaying the expanded text or launching the program depending upon a context of the handwritten user input. *Id.*, para. 48, para. 49, para. 52, Fig. 10, steps 1009, 1007, Fig. 12, steps 1208, 1209.

The invention of claim 37 is directed to the method of claim 1. The method includes determining a number of words in the handwritten user input. *Id.*, para. 51. The method further includes displaying the expanded text if the number of words in the handwritten user input is greater than one, and launching the program if the number of words in the handwritten user input is equal to one. *Id.*, para. 52.

The invention of claim 38 is directed to the method of claim 27. The method includes the feature wherein the context includes a number of words in the handwritten user input. *Id.*, para. 51.

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

37 C.F.R. § 41.37(c)(1)(vi)

Claims 1, 13-15, 17, 19, 20, 24-27, and 34-36 stand rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 6,493,464 to Hawkins et al. (hereinafter “Hawkins”).

Claims 21 and 37-39 stand rejected under 35 U.S.C. § 103 as being unpatentable over Hawkins.

ARGUMENT

37 C.F.R. § 41.37(c)(1)(vii)

Claims 1, 13-15, 17, 19-21, 24-27, and 34-39 are Patentable Over Hawkins.

Independent Claim 1 and Dependent Claim 13

Independent claim 1 recites, among other features, a method, in a computer, comprising the steps of:

receiving user input identifying a symbol, a text expansion, and a program;
associating the text expansion and the program with the symbol; receiving
handwritten user input; determining whether the handwritten user input represents
the symbol; determining a context in which the handwritten user input is written;

choosing between either the text expansion or the program depending upon the determined context; and either displaying the text expansion or launching the program depending upon the outcome of the choosing step.

Hawkins fails to teach or suggest all the features of claim 1. Hawkins describes a pen-based text input system capable of interpreting a special, pre-defined set of single stroke glyphs. *See* Hawkins, Abstract. Hawkins describes a system wherein user input is identified as a pre-determined glyph. *Id.*, col. 7 lines, 26-43. The glyph is then interpreted according to a dictionary that is consulted and, depending on the type of stroke input, an output is provided in response to a stroke that matches the defined glyph. *Id.*, col. 7, lines 44-54.

Hawkins fails to teach or suggest associating a text expansion and a program with a symbol. The final Office Action cites column 12, lines 2-11 and asserts that a user-maintained glossary could be built where the user could define sequences of characters – or symbols, text or program functions – to be associated with a stroke. *See* Final Office Action p. 3. While Hawkins describes associating a stroke with “character(s), symbol(s), text, or program functions,” (Hawkins, col. 12, lines 58-60) Hawkins nevertheless fails to teach or suggest associating the user input with two different expansions: both a text expansion *and* a program.

In fact, Hawkins states that, at any point in time, the glyph dictionary specifies *exactly one interpretation* for each input glyph that the system knows how to recognize. *Id.*, col. 8, lines 35-37 (Emphasis added).

Figures 4A and 4B of Hawkins illustrate that a user input may be associated with more than one *character* – which is consistent with the reference to “character(s)” in col., 12, lines 58-60 (Emphasis added). However, Hawkins still requires that the same stroke to be associated with only one type of expansion – i.e., one or more different characters. The final Office Action attempts to over-extend beyond what is actually disclosed by Hawkins, asserting that Hawkins also associates a stroke with multiple different types of expansions. This is not so; there is simply no teaching or suggestion in Hawkins for such a concept.

Also, while Hawkins describes different associations that may be associated with an input glyph, these are described only as being alternatives: “characters(s), symbol(s), text, *or* program functions.” (*Id.*, col. 12, lines 58-60) (Emphasis added).

Accordingly, Hawkins fails to teach or suggest associating a symbol with both a text expansion and a program, as recited in claim 1. For the same reasons, Hawkins also necessarily fails to teach or suggest choosing between either the text expansion or the program depending upon the determined context, as further recited in claim 1. Although the term “context” is used in Hawkins, the term is used to identify a type of expansion itself. *See* Hawkins col. 12, lines 13-14 and 58-60. For instance, Hawkins states that a “stroke may be associated with any context, such as character(s), symbol(s), text or program functions.” *Id.*, col. 12, lines 58-60. In contrast, as used in claim 1, the term “context” identifies the circumstances surrounding the handwritten user input that is used to determine the type of expansion to display.

For at least these reasons, it is submitted that claim 1 is allowable over Hawkins.

Moreover, all claims depending from claim 1, including claim 13, are also allowable by virtue of depending from an allowable claim.

Dependent Claim 37

Claim 37 is allowable by virtue of depending from allowable claim 1. In addition, claim 37 recites, among other features, “determining a number of words in the handwritten user input and displaying the expanded text if the number of words in the handwritten user input is greater than one, and launching the program if the number of words in the handwritten user input is equal to one.” The final Office Action asserts that Figure 5A of Hawkins illustrates typical end of word indicators, such as “space,” and that such end of word indicators would allow a determination to be made as to whether the word count was equal to one. *See* Final Office Action at p. 9. Respectfully, at best Fig. 5A implies that Hawkins is capable of determining when a handwritten word has ended. However, this does not by itself lead to the conclusion that Hawkins is therefore capable of determining the *number* of words in a user input and performing a particular action *depending upon* that determination, i.e., displaying the expanded text if the number of words is greater than one and launching a program if the number of words is equal to one.

In any event, the final Office Action’s assertion that an alleged prior art system is *capable of* performing a function is irrelevant to the question of obviousness. Any computer can, in

theory, be programmed to perform any function. This does not mean that all computer-related inventions are therefore obvious over existing computers.

Moreover, the alleged motivation for modifying Hawkins is totally unrelated to the proposed modification. The Office Action alleges that it would have been obvious “to use a non-printing character symbol to indicate the user had entered a complete word in order to allow the system to be used for word processing functions.” See Final Office Action at p. 9. Even assuming this as true merely for the sake of argument, this does not address why Hawkins would take that information about where words begin and end and determine the number of words, and then from the number of words perform one of two claimed functions – (a) displaying the expanded text if the number of words in the handwritten user input is greater than one, or (b) launching the program if the number of words in the handwritten user input is equal to one. To the contrary, at best the final Office Action’s proposal would result in the system of Hawkins merely being able to determine the beginning and ending of each word, but not to perform the claimed functions resulting from a count of the number of words.

For at least these reasons, Appellants respectfully assert that claim 37 is allowable.

Independent Claim 14 and Dependent Claims 15, 17, 19-21, 24-26, and 35

Independent claim 14 includes language similar to claim 1 and recites, among other features, a method comprising the steps of “...receiving first handwritten user input including at least first handwritten user input, the first handwritten input being associated with both expanded text and a program.” As discussed above, Hawkins fails to teach or suggest associating a first user input with *both* expanded text and a program. Accordingly, Appellants respectfully assert that claim 14 is allowable.

Moreover, all claims depending from claim 14, including claims 15, 17, 19-21, 24-26, and 35, are also allowable by virtue of depending from an allowable claim.

Independent Claim 27 and Dependent Claims 34, 36, and 39

Independent claim 27 recites, “recognizing the handwritten user input to determine a symbol; determining expanded text represented by the symbol; determining a program

represented by the symbol.” For similar reasons as discussed above, since Hawkins does not teach or suggest associating a stroke with both a character and a program, Hawkins necessarily fails to teach or suggest determining expanded text represented by the symbol *and* determining a program represented by that same symbol. Accordingly, Appellants assert that claim 27 is allowable over Hawkins.

Moreover, all claims depending from claim 27, including claims 34, 36, and 39, are also allowable by virtue of depending from an allowable claim.

Dependent Claim 38

Claim 38 depends from claim 27 and is allowable for at least the same reasons discussed with respect to claim 27. Moreover, claim 27 recites, among other features, “...either displaying the expanded text or launching the program depending upon a context of the handwritten user input.” Claim 38 recites the additional feature, “wherein the context includes a number of words in the handwritten user input.” While Hawkins does describe symbols that may be used to indicate the end of a word, Hawkins fails to teach or suggest determining a number of words in the input and displaying expanded text or launching a program based on that number of words. Again, merely proposing that Hawkins is capable of counting words, is totally disconnected with what is done as a result of that count as required by claims 27 and 38 – that the context, upon which either expanded text is displayed or a program is launched, includes the number of words. Accordingly, Appellants respectfully assert that claim 38 is allowable for at least these reasons.

CONCLUSION

For all of the foregoing reasons, Appellant respectfully submits that the final rejection of claims 1, 13-15, 17, 19-21, 24-27, and 34-39 is improper and should be reversed.

Respectfully submitted,
BANNER & WITCOFF, LTD.

Dated: May 18, 2007

By: /Elizabeth A. Almeter/
Elizabeth A. Almeter
Registration No. 57,019

1100 13th Street, N.W.
Suite 1200
Washington, D.C. 20005
Tel: (202) 824-3000
Fax: (202) 824-3001

CLAIMS APPENDIX

37 C.F.R. § 41.37(c)(1)(viii)

Claims involved in the appeal:

1. In a computer, a method comprising steps of:

receiving user input identifying a symbol, a text expansion, and a program;

associating the text expansion and the program with the symbol;

receiving handwritten user input;

determining whether the handwritten user input represents the symbol;

determining a context in which the handwritten user input is written;

choosing between either the text expansion or the program depending upon the determined context; and

either displaying the text expansion or launching the program depending upon the outcome of the choosing step

.

13. A computer-readable medium storing computer-executable instructions for performing the steps recited in claim 1.

14. In a computer, a method comprising steps of:

receiving first handwritten user input including at least first handwritten user input, the first handwritten input being associated with both expanded text and a program;

determining whether the first handwritten user input includes second handwritten user input in addition to the first handwritten user input;

choosing between either the expanded text or the program depending upon whether the first handwritten user input includes the second handwritten user input; and

in response to the first handwritten user input, either displaying the expanded text or launching the program, depending upon the outcome of the choosing step.

15. The method of claim 14, wherein the first handwritten user input consists of a single word.

17. The method of claim 14, further including:

comparing the first handwritten user input with a predetermined set of symbols;
and
based on the step of comparing, determining either the expanded text or the program.

19. The method of claim 14, wherein the second handwritten user input includes any handwritten user input other than the first handwritten user input that is simultaneously displayed with the first handwritten user input.

20. The method of claim 14, wherein the second handwritten user input consists of any handwritten user input on a same line as the first handwritten user input and simultaneously displayed with the first handwritten user input.

21. The method of claim 14, wherein the step of choosing includes determining whether a total handwritten user input word count is equal to one, and if so, then determining that the first handwritten user input does not include the second handwritten input.

24. The method of claim 14, further including third determining whether all handwritten user input has stopped, the step of choosing being performed in response to determining that all handwritten user input has stopped.

25. The method of claim 14, further including a step of waiting a predetermined period of non-zero time after the step of receiving, the step of choosing being performed after the step of waiting.

26. A computer-readable medium storing computer-executable instructions for performing the steps recited in claim 14.

27. In a computer, a method comprising steps of:
receiving handwritten user input;
recognizing the handwritten user input to determine a symbol;
determining expanded text represented by the symbol;
determining a program represented by the symbol; and
either displaying the expanded text or launching the program depending upon a context of the handwritten user input.

34. A computer-readable medium storing computer-executable instructions for performing the steps recited in claim 27.

35. The method of claim 14, further including:
prior to receiving the first handwritten user input, receiving user input identifying the text expansion and the program.

36. The method of claim 27, further including, prior to receiving the handwritten user input, receiving user input identifying the symbol, the text expansion, and the program.

37. The method of claim 1, wherein the method includes:
determining a number of words in the handwritten user input; and

displaying the expanded text if the number of words in the handwritten user input is greater than one, and launching the program if the number of words in the handwritten user input is equal to one.

38. The method of claim 27, wherein the context includes a number of words in the handwritten user input.

39. The method of claim 27, wherein the method includes:

determining a number of words in the handwritten user input; and

displaying the expanded text if the number of words in the handwritten user input is greater than one, and launching the program if the number of words in the handwritten user input is equal to one.

EVIDENCE APPENDIX

37 C.F.R. § 41.37(c)(1)(ix)

None

RELATED PROCEEDINGS APPENDIX

37 C.F.R. § 41.37(c)(1)(x)

None